

---

Subject: [PATCH][ just for review] memory controller enhancements [5/5]  
memory.stat file

Posted by [KAMEZAWA Hiroyuki](#) on Thu, 11 Oct 2007 09:54:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Show accounted information of memory cgroup by memory.stat file

Signed-off-by: YAMAMOTO Takashi <yamamoto@valinux.co.jp>

Signed-off-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

mm/memcontrol.c | 54 +++  
1 file changed, 54 insertions(+)

Index: devel-2.6.23-rc8-mm2/mm/memcontrol.c

=====

--- devel-2.6.23-rc8-mm2.orig/mm/memcontrol.c

+++ devel-2.6.23-rc8-mm2/mm/memcontrol.c

@@ -28,6 +28,7 @@

#include <linux/swap.h>

#include <linux/spinlock.h>

#include <linux/fs.h>

+#include <linux/seq\_file.h>

#include <asm/uaccess.h>

@@ -828,6 +829,55 @@ static ssize\_t mem\_force\_empty\_read(stru  
{

+static const struct mem\_cgroup\_stat\_desc {

+ const char \*msg;

+ u64 unit;

+} mem\_cgroup\_stat\_desc[] = {

+ [MEM\_CGROUP\_STAT\_PAGECACHE] = { "page\_cache", PAGE\_SIZE, },

+ [MEM\_CGROUP\_STAT\_RSS] = { "rss", PAGE\_SIZE, },

+ [MEM\_CGROUP\_STAT\_CHARGE] = { "charge", PAGE\_SIZE, },

+ [MEM\_CGROUP\_STAT\_UNCHARGE] = { "uncharge", PAGE\_SIZE, },

+ [MEM\_CGROUP\_STAT\_ACTIVE] = { "active", PAGE\_SIZE, },

+ [MEM\_CGROUP\_STAT\_INACTIVE] = { "inactive", PAGE\_SIZE, },

+};

+

+static int mem\_control\_stat\_show(struct seq\_file \*m, void \*arg)

+{

+ struct cgroup \*cont = m->private;

+ struct mem\_cgroup \*mem\_cont = mem\_cgroup\_from\_cont(cont);

+ struct mem\_cgroup\_stat \*stat = &mem\_cont->stat;

+ int i;

+

```

+ for (i = 0; i < ARRAY_SIZE(stat->cpustat[0].count); i++) {
+   unsigned int cpu;
+   s64 val;
+
+   val = 0;
+   for (cpu = 0; cpu < NR_CPUS; cpu++)
+     val += stat->cpustat[cpu].count[i];
+   val *= mem_cgroup_stat_desc[i].unit;
+   seq_printf(m, "%s %lld\n", mem_cgroup_stat_desc[i].msg, val);
+ }
+ return 0;
+}
+
+static const struct file_operations mem_control_stat_file_operations = {
+ .read = seq_read,
+ .llseek = seq_lseek,
+ .release = single_release,
+};
+
+static int mem_control_stat_open(struct inode *unused, struct file *file)
+{
+ /* XXX __d_cont */
+ struct cgroup *cont = file->f_dentry->d_parent->d_fsdata;
+
+ file->f_op = &mem_control_stat_file_operations;
+ return single_open(file, mem_control_stat_show, cont);
+}
+
+
+
+static struct cftype mem_cgroup_files[] = {
+ {
+ .name = "usage_in_bytes",
@@ -855,6 +905,10 @@ static struct cftype mem_cgroup_files[]
+ .write = mem_force_empty_write,
+ .read = mem_force_empty_read,
+ },
+ {
+ .name = "stat",
+ .open = mem_control_stat_open,
+ },
+ };
+
+static struct mem_cgroup init_mem_cgroup;

```

---

Containers mailing list  
Containers@lists.linux-foundation.org

